COMPUTER SCIENCE & INFORMATION TECHNOLOGY

Theory of Computation

Comprehensive Theory
with Solved Examples *and* Practice Questions

MADE EASY PUBLICATIONS

www.madeeasypublications.org

MADE EASY Publications Pvt. Ltd.

Corporate Office: 44-A/4, Kalu Sarai (Near Hauz Khas Metro Station), New Delhi-110016 | **Ph. :** 9021300500

Email : infomep@madeeasy.in | **Web :** www.madeeasypublications.org

Theory of Computation

EDITIONS

First Edition : 2015
Second Edition : 2016
Third Edition : 2017
Fourth Edition : 2018
Fifth Edition : 2019
Sixth Edition : 2020
Seventh Edition : 2021
Eighth Edition : 2022
Ninth Edition : 2023

Tenth Edition : 2024

CONTENTS

Theory of Computation

Theory of Computation

GOAL OF THE SUBJECT

Theory of Computation deals with automata theory and formal languages. It is the study of "*Abstract Model of Computation*". This subject helps to solve various problems using the following modes.

- Finite Automata
- Push Down Automata
- Linear Bound Automata
- Turing Machine

It is very important for every computer programmer to know what are the problems that can be solved and what cannot be solved. This subject explains the capabilities and limitations of computation.

INTRODUCTION

To understand the formal model of computation, this book provides you rigorous presentation of concepts, models, examples accompanied by practice problems and student assignment.

This book is organized with all models that makes it easier to read, understand and acquire quick interpretation of all models. It uses definitions and properties of mathematical models to explain the capabilities and limitations of Problems/Computation/Programs/Languages.

The following information provides a brief description of the chapters in this book.

Chapter 1 (Grammars, Languages & Automata): This chapter deals with all notations, and definitions of theory of computation such as Grammar, Equivalence of Language, Chomsky Hierarchy Model and languages with their relationships in the computation model.

Chapter 2 (Regular Languages & Finite Automata): This chapter explains in detail about type-3 formal languages acceptance by Finite Automata with representations like Regular Expressions, Regular Sets, and Regular Grammars. It also covers the Closure Properties and Decision Properties of regular languages with detailed proofs and Moore and Mealy machines.

Chapter 3 (Context Free Languages & Push Down Automata): This chapter covers type-2 formal languages acceptance by Push Down Automata and representations with equivalences between CFG and CFL, Closure Properties and Decision Properties of context free languages with detailed proofs and identify the CFL and non-CFL languages.

Chapter 4 (REC, RE Languages & Turing Machines, Decidability): This chapter deals with Turing Machine Decidable and Undecidable problems of formal languages and automata theory and Closure Properties and Decision Properties of context free languages with detailed proofs.

■■■■

CHAPTER

1

Grammars, Languages and Automata

1.1 INTRODUCTION

1.1.1 Alphabet(Σ)

Definition: An Alphabet is a finite non-empty set of symbols.

- $\Sigma = \{a, b\}$ is alphabet with 2 symbols *a* and *b* and hence binary alphabet.
- $\Sigma = \{0, 1\}$ is a binary alphabet.
- $\Sigma = \{0, 1, 2, \ldots 9\}$ is decimal alphabet.
- '∊' [epsilon] cannot be used as alphabet symbol because '∊' is a special symbol used to denote null string.
- Symbols in alphabet are atomic things.

Example 1.1 Which of the following is not an alphabet?

(a) {0, 1} (b) { }

(c) $\{x\}$ (d) $\{a\}$

(e) {∊} (f) {0, 1, ... 01}

(g) $\{a, b, c, \ldots\}$

Solution :

1. {∊} is not an alphabet since ∊ is special symbol.
2. { } is not an alphabet since empty set.
3. $\{a, b, c, \ldots\}$ is not an alphabet since infinite set.
4. {0, 1, 01} is not an alphabet since 0 is proper subset of 01 (atomic things).

1.1.2 String

Definition: A string is any sequence of zero or more finite number of symbols over the given alphabet Σ.

"*abb*" is the string over $\Sigma = \{a, b\}$. Every symbol is a string of length 1.

Empty string (ε or λ): Empty string is a string with zero number of symbols in the sequence. Empty string is also called as "Null String".

1.1.3 Operations on Strings

1. **Length of a string:** The number of symbols in the sequence of given string is called "length of a string"
   - Length of string *abb* = $|abb|$ = 3, where $\Sigma = \{a, b\}$
   - $|\varepsilon| = 0$. The length of empty string is zero.
   - $|a| = 1$, $|ab| = 2$

Example 1.2 How many strings are possible with alphabets $\{a, b\}$ of length n?

Solution :

$$\text{Number of string} = 2 \times 2 \times ..... 2 = 2^n$$

Where each '2' represents the number of choices possible i.e. either *a* or *b*.

Example 1.3 How many strings possible with alphabet $\Sigma = \{a, b, c\}$ upto length n?

Solution :

Number of strings = Number of string of length 0 + Number of string of length 1 + ... Number of string of length n

$$\Rightarrow \quad |\Sigma|^0 + |\Sigma|^1 ..... |\Sigma|^n$$

$$\Rightarrow \quad |\Sigma|^0 \left[\frac{|\Sigma|^{n+1} - 1}{|\Sigma| - 1}\right]$$

$$3^0 \left[\frac{3^{n+1} - 1}{3 - 1}\right] = \frac{3^{n+1} - 1}{2}$$

2. **Substring of a string:** A sequence of symbols from any part of the given string over an alphabet is called a "Substring of a string" OR zero or more consecutive symbols of a string.

   For string abb over $\Sigma = \{a, b\}$. The possible substrings are:
   - Zero length substring: ε
   - One length substrings: *a*, *b*
   - Two length substrings: *ab*, *bb*
   - Three length substrings: *abb*

   Mathematically, substring(w) = $\{y \mid w = xyz\}$

   "Proper Substring" is a substring and its length is less than given string length. The string "*abb*" is not a proper substring of the string "*abb*" but is a substring of "*abb*".

3. **Prefix of a string:** A substring with the sequence of beginning symbols of a given string is called a "Prefix".

   For string "*abb*", the possible prefixes of *abb* are:
   - ε, (zero length prefix)

Student's Assignments 1

Q.1 Finite automata has
(a) Finite Control (b) Read only head
(c) Both (a) and (b) (d) None of these

Q.2 Compared to NFA, DFA has
(a) Less power
(b) More power
(c) Deterministic transition
(d) None of these

Q.3 Finite Automata is used in which of the following?
(a) Pattern matching
(b) Sequential circuit design
(c) Compiler design
(d) All (a), (b) and (c)

Q.4 A finite automata is
(a) Acceptor that accepts a regular language
(b) Transducer that computes some simple functions
(c) Both (a) and (b)
(d) None of these

Q.5 The language accepted by finite automata is
(a) Context-free (b) Regular
(c) Non-regular (d) None of these

Q.6 In computer system, finite automata program can be stored using
(a) Table
(b) Two-dimensional array
(c) Graph
(d) All (a), (b) and (c)

Q.7 Which of the followings is true
(a) All NFAs are DFAs
(b) All DFAs are NFAs
(c) Both (a) and (b)
(d) NFA and DFA have different power

Q.8 Two-way finite automata has
(a) Two tapes
(b) Two heads
(c) Bi-directional head movement
(d) All the above

Q.9 The behavior of a NFA can be simulated by a DFA
(a) Always (b) Sometime
(c) Never (d) Depend on NFA

Q.10 A FA with deterministic transitions capability is known as
(a) NFA (b) DFA
(c) 2DFA (d) NFA with ε-moves

Answer Key:

1. (c)	**2.** (c)	**3.** (d)	**4.** (c)	**5.** (b)
6. (d)	**7.** (b)	**8.** (c)	**9.** (a)	**10.** (b)

Student's Assignments 1 Explanations

1. (c)

Finite automata has finite control, read only head, tape is unbounded but stores finite length string.
∴ Both (a) and (b) are correct.

2. (c)

NFA and DFA both accepts regular language.
NFA and DFA are equivalent.
But NFA is non-deterministic whereas DFA is deterministic.
∴ Option (c) is correct.

3. (d)

FA can be used in pattern matching, sequential circuit design and lexical analysis of compiler design, etc.
∴ Option (d) is correct.

4. (c)

A finite automata may be an acceptor or a transducer.
∴ Option (c) is correct.

5. (b)

FA accepts regular languages
∴ Option (b) is correct.

6. (d)

In computer, FA program can be stored using table, graph, array, etc.
∴ Option (d) is correct.

7. (b)

Every DFA is a special case of NFA, but every NFA need not be a DFA.

∴ Option (b) is correct.

8. (c)

Two-way finite automata has bi-directional head. The head can move in both directions: left or right. This machine is called as 2DFA or 2NFA based on nature of acceptance.

∴ Option (c) is correct.

9. (a)

NFA and DFA are equivalent.

Hence NFA always simulated by a DFA and NFA can be converted to a DFA

∴ Option (a) is correct.

10. (b)

FA with deterministic transitions capability is called DFA.

∴ Option (b) is correct.

Student's Assignments 2

Q.1 A FA with non-deterministic transitions capability is known as _______.
(a) NFA (b) DFA
(c) 2DFA (d) NFA with ε-moves

Q.2 A FA having more than one input tape has _______
(a) More power than one tape FA
(b) Equal power as one tape FA
(c) Less power than one tape FA
(d) None of these

Q.3 A FA having more than one reading head has _______.
(a) More power than one reading head FA
(b) Equal power as one reading head FA
(c) More storage than one reading head FA
(d) Both (b) and (c)

Q.4 Limitation of a FA is _______
(a) Writing on tape
(b) Finite memory
(c) pattern recognition
(d) Both (a) and (b)

Q.5 What is true about finite automata? It is
(a) Acceptor (b) Recognizer
(c) Pattern Matcher (d) All of these

Q.6 Which is false?
(a) In DFA, there is only one transition for every input symbol from any state.
(b) In NFA, there is zero or more transition for an input symbol from any state.
(c) All DFAs are NFAs, but not all NFA are DFA. So, NFA have more power as compare to DFA.
(d) The language accepted by NFA is equivalent to a DFA.

Q.7 A finite automata can
(a) Check the validity of input string
(b) Manipulate the input and calculate the output
(c) Both (a) and (b)
(d) None of these

Q.8 FSM can recognize
(a) Any language
(b) Only regular language
(c) Only context-free
(d) None of these

Q.9 An automaton is a _________ device and a grammar is a _________ device.
(a) generative, cognitive
(b) generative, acceptor
(c) acceptor, cognitive
(d) cognitive, generative

Q.10 Context-free language can be recognized by
(a) finite state automaton
(b) linear bounded automata
(c) push down automata
(d) both (b) and (c) above

Q.11 Can a DFA simulate NFA?
(a) no (b) yes
(c) some time (d) depends on NFA

Q.12 A language L is accepted by a finite automaton if and only if it is
(a) context-free
(b) context-sensitive
(c) recursive
(d) expressible by a right-linear grammar

Q.13 The basic limitation of FSM is that
(a) it can not remember arbitrary information
(b) it sometimes fails to recognize regular language
(c) it cannot be used in pattern recognization
(d) all of these

Q.14 Regarding the power of recognizing the languages, which of the following statements is false?
(a) The NDFA are equivalent to DFA
(b) NPDA are equivalent to DPDA
(c) NDTMs are equivalent to DTMs
(d) Universal TM is equivalent to standard TM

Q.15 Which of the following is false?
(a) The languages accepted by FAs are regular languages
(b) Every DFA is an NFA
(c) There are some NFAs for which no DFA can be constructed
(d) If L is accepted by an NFA with ε-transitions then L is also accepted by an NFA without ε transition

Q.16 The languages accepted by DPDA are _______.
(a) CFLs (b) CSLs
(c) DCFLs (d) RELs

Q.17 The languages accepted by PDA are _______
(a) CFLs (b) CSLs
(c) DCFLs (d) RELs

Q.18 The languages accepted by NPDA are _______
(a) CFLs (b) CSLs
(c) DCFLs (d) RELs

Q.19 The languages accepted by LBA are _______
(a) CFLs (b) CSLs
(c) DCFLs (d) RELs

Q.20 The languages accepted by TMs are _______
(a) CFLs (b) CSLs
(c) DCFLs (d) RELs

Q.21 Find the correct relation of expressive power of machines
(a) FA < PDA < LBA < TM
(b) FA > PDA > LBA > TM
(c) TM < PDA < LBA < FA
(d) TM > PDA > LBA > FA

Q.22 A language L is accepted by a PDA if and only if it is
(a) Regular language
(b) CSL
(c) Expressible by CFG
(d) Expressible by Right-linear grammar

Q.23 A language L is accepted by a TM if and only if it is _______.
(a) Recursive language
(b) Enumerable in lexicographical order
(c) Expressible by unrestricted grammar
(d) None of these

Q.24 Find the correct chomsky hierarchy
(a) Type0 $\subseteq$ Type1 $\subseteq$ Type2 $\subseteq$ Type3
(b) Type3 $\subseteq$ Type1 $\subseteq$ Type2 $\subseteq$ Type0
(c) Type0 $\subseteq$ Type2 $\subseteq$ Type1 $\subseteq$ Type3
(d) None of these

Q.25 Which one of the following is correct for the input alphabet Σ?
(a) $\Sigma^* = \Sigma$ (b) $\Sigma^* = \Sigma^+ - \{\varepsilon\}$
(c) $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$ (d) $\Sigma^* = \Sigma^+ \cap \{\varepsilon\}$

Answer Key:

1. (a)	**2.** (b)	**3.** (b)	**4.** (d)	**5.** (d)
6. (c)	**7.** (c)	**8.** (b)	**9.** (d)	**10.** (d)
11. (b)	**12.** (d)	**13.** (a)	**14.** (b)	**15.** (c)
16. (c)	**17.** (a)	**18.** (a)	**19.** (b)	**20.** (d)
21. (a)	**22.** (c)	**23.** (c)	**24.** (d)	**25.** (c)

Student's Assignments Explanations

1. (a)

Any FA having non deterministic transition capability is known as NFA.

2. (b)

A FA having any number of input tape does not increase its power. Writing capability and two side measurement may increase.
Hence, option (b) is correct.

4. (d)

FA can be used for pattern recognization is compiler design in a lexical analysis phase. Although it does not have capability to write on the tape and because of finite number of states can only acts as an counter.

5. (d)

FA can acts an acceptor, pattern matter and recognizer all.

6. (c)

Expressive power of non-deterministic Turing Machine is same as deterministic Turing Machine.

7. (c)

A finite automation can check the validity of input string by checking whether the FA going to final state or not.
Moore and Mealy machine are the application of finite automata used to generate output based on some input.

8. (b)

Finite state machine can recognize only regular language.

9. (d)

An automation is a cognitive (acceptor) device and a grammar is generative device.

10. (d)

Context-free language can be recognized by any device whose power is more than POA thus both linear bounded automata and POA can recognize CFL.

11. (b)

All DFA's are NFA's. Thus, any DFA can similar NFA.

12. (d)

Right linear grammar generates regular language any finite automation can only accept regular language.

13. (a)

Basic limitation of FSM is that it cannot remember any arbitrary information.

14. (b)

Non-deterministic push down automata has more power than deterministic push down automata.

15. (c)

Expressive power of: DFA ≅ NFA ≅ NFA with ε-moves and they are inter convertible and thus option (c) is false.

16. (c)

Deterministic push down automata is used to accept deterministic context free language.

17. (a)

Push down automata is used to accept context free languages and deterministic context free languages but CFLs are more stricter.

18. (a)

Non deterministic push down automata is used to accept context free language but not deterministic context free language.

19. (b)

Linear bounded automata can be used to accept CFL's, DCFL's, CSL's but if we gives answer is stricter from then CSL's.

20. (d)

TM can accept any of the given language. But strictly RELS. Any other language come under this.

CHAPTER

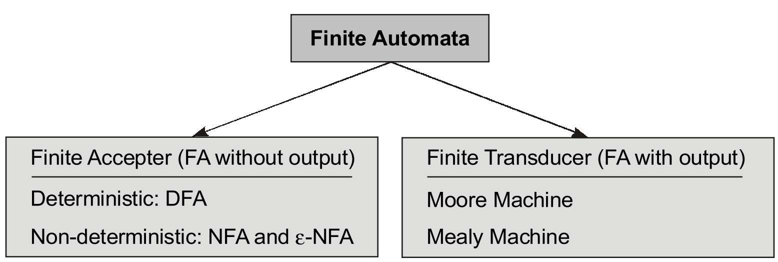
2

Regular Languages and Finite Automata

2.1 INTRODUCTION

- Finite Automaton (FA) is also called as Finite State machine (FSM).
- Finite Automaton has no temporary storage hence it can remember finite information with the help of finite control which contain states and transitions.

2.1.1 Types of Finite Automata

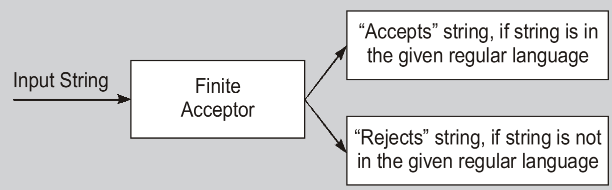


Finite automata is categorized into two types:

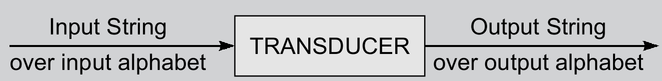
1. **Finite automata with output:** This is of two types Mealy machines and Moore machines.
   (a) In Mealy machine the output is associated with transition (state and input symbol).
   (b) In the Moore machine output is associated with only state.
2. **Finite automata without output:** This is of two types deterministic and non-deterministic.
   (a) **Deterministic Finite Automata (DFA):** It is deterministic finite automata. For every input symbol in DFA there is an exactly one transition from any state of finite automata. It accepts the string by halting at a final state and rejects the string by halting at a non-final state.
   (b) **Non-deterministic Finite Automata (NFA):** It is non deterministic finite automata. For every input symbol in NFA there is zero or more transitions from any state of finite automata. It accepts the string by halting at a final state and rejects the string either by halting at a non-final state or by halting in a dead configuration.
   (c) **ε-NFA:** It is a non deterministic finite automata which can include ε transitions. It has capability of changing the state without reading the input symbol.

Remember

- Acceptor is a machine which accepts or recognizes a language
- Transducer is a machine which produces an output
- Deterministic automaton is one in which each move is uniquely determined by reading an input.
- Non-deterministic automaton is one in which each move may have set of possible actions.
- Moore and Mealy machines are deterministic.
- Finite Automaton accepts a regular language as follows:

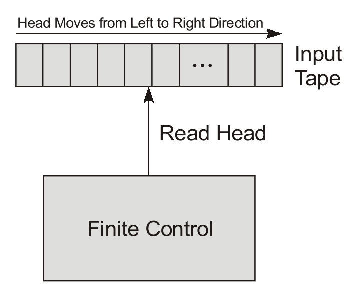


- Finite transducer produces an output for the given input as follows:



2.1.2 Model (Configuration) of Finite Automata

- Input tape contains a string over the given input alphabet.
- Read head is a pointer which reads one input symbol at a time and moves to the next symbol.
- Finite control contains set of finite states and transition functions to take an action based on current state and input symbol scanned.
- For every string the finite control starts from an initial state. If the string is valid, finite control reaches to the final state.



2.1.3 Finite Automata Acceptance

- Finite Automata accepts type-3 formal languages called as regular languages.
- Regular language is either finite or infinite language.
- A language is finite if it contains finite number of strings otherwise it is infinite language.
- Every finite language is regular hence it is possible to construct finite automata to accept the finite language.
- For infinite language the construction of finite automata may or may not possible. If the finite automata does not require infinite memory then it is possible to construct otherwise it is not possible to construct as it requires infinite number of states, which is not allowed in FA.

2.1.4 Regular Language

- All finite languages are regular.
- If the language is infinite over unary alphabet and whose string lengths are forming an arithmetic progression then it is regular language, otherwise it is non-regular.
- If the language is infinite over two or more input symbols and which do not contain any dependency between symbols then the language is regular.

2.1.5 Representations of Regular Language

- Finite Automata (by construction)
- Regular Grammars (by writing productions)
- Regular Sets (by representation of a set)
- Regular Expression (by writing an expression)

2.1.6 Equivalence of Finite Automata

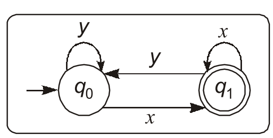
- DFA ≅ NFA ≅ ε-NFA
  All these three finite automata accept same class of languages called as regular languages
- Every DFA is also a special case of NFA
- NFA can be converted to DFA using subset construction
- ε-NFA can be converted to NFA and also to DFA

2.1.7 Representations of FA

FA = (Q, Σ, δ, q_0, F) is a 5-tuple notation

where, Q is set of finite states,
Σ is an input alphabet contains finite number of input symbols.
δ is transition function defined over transitions of FA for state and input.
q_0 is initial state or start state of FA, $q_0 \in Q$.
F is set of final states of FA.

1. State Transition Diagram:
   (a) Every state represented by a circle.
   (b) For an Initial state, a circle is associated with an arrow.
   (c) Final state represented by double circle.
   (d) Every state is connected with transitions associated with an input.
   (e) In the following diagram: q_0 and q_1 are states, q_0 is an initial state and q_1 is a final state.



Q = {q_0, q_1}
Σ = {x, y}
q_0 is initial state
q_1 is a final state

2. State transition table:
   (a) Table contains rows and columns.
   (b) Each row corresponds to the state of DFA.
   (c) Each column corresponds to the input symbol of DFA.
   (d) Initial state is associated with an arrow.
   (e) Final state is associated with a circle or star.
   (f) In the following table: q_0 and q_1 are states, q_0 is an initial state and q_1 is a final state.

	x	y
→ q_0	q_1	q_0
*q_1 or (q_1)	q_1	q_0

2.2 DETERMINISTIC FINITE AUTOMATA (DFA)

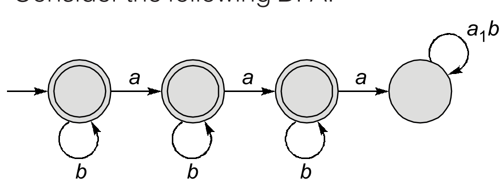
- If for every input symbol of an alphabet there is an exactly one transition from every state of finite automata then such FA is called as DFA.

Student's Assignments 1

Q.1 Consider a regular expression $R = (a + \varepsilon)(bb^*a)^*$. What is the language generated by R over $\Sigma = \{a, b\}$
(a) Set of all strings that do not contain aa.
(b) Set of all strings that do not contain two or more consecutive a's.
(c) Set of all strings that do not end with b and do not contain two or more consecutive a's.
(d) None of these

Q.2 Let $L = \{pq \mid p, q \in \{0, 1\}^*$ and dec(p) × dec(q) = 10 (ten)}, where dec(x) is decimal equivalent of binary number x. Find the language L by assuming that leading zero's are allowed for the strings of L.
[01 = 001 = 0001]
(a) Regular
(b) CFL but not regular
(c) Recursive but not CFL
(d) None of these

Q.3 Consider the following DFA:



Identify the language accepted by the DFA?
(a) All strings not contain aaa
(b) All strings not starting with aaa.
(c) All strings with no more than two a's
(d) None of these

Q.4 Let L_1 and L_2 be languages over Σ. If L_1 is finite language and $L_1 \cup L_2$ is regular then L_2 is ______?
(a) Regular language and finite
(b) Regular language and infinite
(c) Need not be regular
(d) None of these

Q.5 Find the number of states in minimized DFA for the regular expression $a^+b^+ + b^+a^+$
(a) 4
(b) 5
(c) 6
(d) None of these

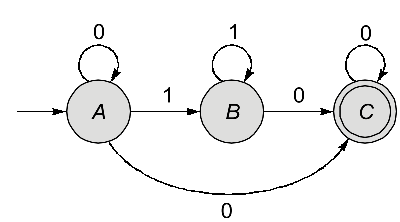
Q.6 Consider the following configuration of a DFA. DFA = $(Q, \Sigma, \delta, q_0, A)$ where $Q = \{q_0, q_1\}$, q_0 is initial state, A is set of final states.
Given, $A = \{q_1\}$ and $\delta(q_i, x) = q_{1-i}$, $x \in \Sigma$, $i = 0, 1$.
Identify the language accepted by the above DFA?
(a) Set of all strings over a given alphabet Σ
(b) Set of all strings of odd length over Σ
(c) Set of all strings of even length over Σ
(d) None of these

Q.7 Identify the non regular language from the following.
(a) $\{x \mid x \in (0+1)^*,\ |x|$ is odd, First symbol of x is $1\}$
(b) $\{x \mid x \in (0+1)^*, |x|$ is odd, Middle symbol of x is $1\}$
(c) $\{x \mid x \in (0+1)^*,\ |x|$ is odd, Last symbol of x is $1\}$
(d) None of these

Q.8 Consider a regular language R over $\Sigma = \{a, b\}$ which is equivalent to the regular expression $(ab + b)^*$. How many equivalence classes of Σ^* are present for the language R?
(a) 2
(b) 3
(c) 4
(d) 5

Q.9 Let $L = \{w\,x\,w^R \mid w \in (a + b)^*, x \in (a + b)\}$. The complement of language L is ________.
(a) Regular
(b) Finite language
(c) Non regular language
(d) None of these

Q.10 What is the language accepted by the following NFA?



(a) $(0 + 1)^*0$
(b) $0^*\,1^*\,0^*$
(c) $0^*\,1^*\,0^*\,0$
(d) $(0 + 1)^+$

Q.11 Find the number of states in minimized DFA that accepts a language over $\Sigma = \{a, b\}$ where each string has exactly three a's and atleast two b's.

(a) 12 (b) 13
(c) 14 (d) 15

Q.12 Let $L_1 = a^*b^*$ and $L_2 = \{ab\}$. L_3 = Prefix $(L_1^* \cap L_2)$, where prefix (L) = $\{u \mid uv \in L$ for any $v\}$. Find the number of strings in L_3?

(a) 3 (b) 4
(c) 5 (d) 6

Q.13 How many number of states are there in the minimized DFA that accepts the following language L.

$$L = \{a^{3(n+1)} \mid n \geq 0\}$$

(a) 3 (b) 4
(c) 5 (d) 6

Q.14 Find the number of states in DFA which accepts a language such that each block of 4 consecutive symbols of every string contain at least two a's for $\Sigma = \{a, b\}$. If string length is less than 4 then it must be accepted.

(a) 10 (b) 11
(c) 12 (d) 13

Common Data Questions (15 and 16):

Consider the following regular expression R.

$$R = a^*b^* + b^*a^*$$

Q.15 How many final states exist in the minimized DFA that accepts a language equivalent to R.

(a) 3 (b) 4
(c) 5 (d) 6

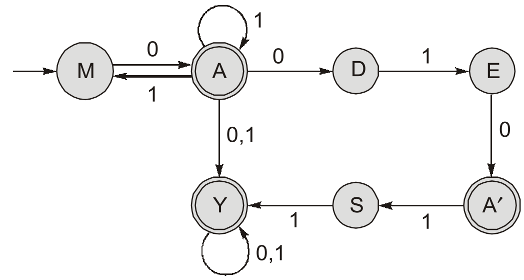
Q.16 How many equivalence classes of Σ^* to represent a language which is equivalent to R.

(a) 3 (b) 4
(c) 5 (d) 6

Q.17 Find the number of states in DFA that recognizes the language where all strings that do not contain the substring bab, for $\Sigma = \{a, b\}$

(a) 3 (b) 4
(c) 5 (d) 6

Q.18 Find the language accepted by the following NFA.



(a) All strings of 0's and 1's
(b) All strings starting with '1'
(c) All strings starting with '0'
(d) None of these

Q.19 In MADE EASY, "Reddy" found a regular language named as L1. "Kumar" is a friend of Reddy who found a new language L2. "Singh" is teaching TOC to Reddy and Kumar, He has collected both L1 and L2 to compute the new language L3. If L3 = L1 – L2 then Singh found that L3 is _______.

(a) Regular language
(b) Non regular language
(c) Finite language
(d) Need not be regular language

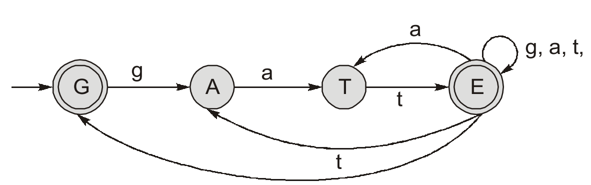
Q.20 Let L be the language formed by tossing a coin. Each string in L has n-length and the sequence in each string depends on the result of 'n' tosses. Each toss can result in either Head (H) or Tail (T). Find the number of strings in L over the input alphabet $\Sigma = \{T, H\}$

(a) n (b) n^2
(c) 2^n (d) n^n

Q.21 How many states are required to construct a DFA that accepts a binary language 'L', where every string in 'L' contains a substring '0' or '1'.

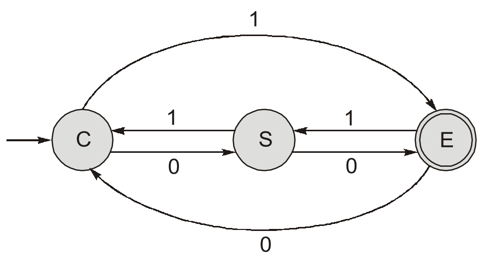
(a) 1 (b) 2
(c) 3 (d) 4

Q.22 Consider the following NFA for $\Sigma = \{g, a, t, e\}$



Find the number of states in equivalent DFA.
(a) 4 (b) 5
(c) 6 (d) 8

Q.23 Consider the following finite automata F1 that accepts a language 'L'



Let F2 be a finite automata which is obtained by the reversal of F1. Then which of the following is correct?
(a) L(F1) ≠ L(F2) (b) L(F1) = L(F2)
(c) L(F1) ⊆ L(F2) (d) L(F2) ⊆ L(F1)

Q.24 Let $L_1 = (0 + 1)^*$, $L_2 = (01^*0 + 1^*)$ and $L_3 = (01^+0 + 1^+ + \varepsilon)$. If $L = L_1 \cap L_2 \cap L_3$ then find number of strings in L which do not contain 11.
(a) 2 (b) 3
(c) 4 (d) 5

Q.25 Let $L = \{w | w \in (a + b)^*$ and w does not end in $ba\}$. Find the number of states needed for minimal DFA to accept L
(a) 2 (b) 3
(c) 4 (d) 5

Q.26 Let L = {madeeasy} over Σ = {m, a, d, e, s, y}, L1 = prefix(L) and L2 = L1/Σ*. Find the number of strings in L2. Assume L1 and L2 do not include empty string.
(a) 6 (b) 7
(c) 8 (d) 9

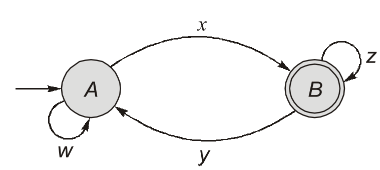
Q.27 Let L1 be the language accepted by DFA D1 and L2 be the language accepted by DFA D2. Similarly L3 and L4 languages are accepted by NFA N1 and N2 respectively.
1. D2 is obtained by swapping the accepting and non-accepting states of D1
2. N2 is obtained by swapping the accepting and non-accepting states of N1

Which of the following statement is incorrect?
(a) L2 is complement of L1
(b) Class of languages recognized by NFA's is not closed under complement
(c) Class of languages recognized by DFA's is closed under complement
(d) None of these

Q.28 Assume every L_i is regular language for $i \geq 1$. If infinite union is defined as : $X = L_1 \cup L_2 \cup L_3 \ldots = \bigcup_{i \geq 1} L_i$ then X is ______.
(a) Regular
(b) Finite language
(c) Non regular language
(b) Need not be regular language

Q.29 Let L_1 be a regular language and $L_2 = L_1^*$. Then L_2 is ______.
(a) Always finite language
(b) Always infinite language and regular
(c) Need not be infinite language
(d) None of these.

Q.30 Consider the following state diagram:



Assume $w, x, y, z \in \{0, 1\}$. If above state diagram is possible to make a DFA with deterministic inputs w, x, y, z then find number of possible DFA's. [Assume all given states are fixed]
(a) 4 (b) 5
(c) 6 (d) 7

Q.31 Let L be the language over {a,b} and it contain all strings with following rules.
1. Every string starts with "a"
2. Every string contains "ab"
3. Every string ends with "b"

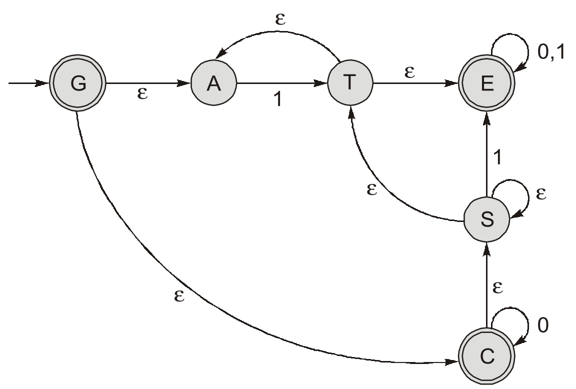
Find the number of states required in the minimized DFA that accepts L.
(a) 4 (b) 5
(c) 6 (d) 7

Q.32 Let L be a language and $L = \{w \mid w$ has exactly one 'b', $w \in \{a, b\}^*$, $|w|$ is even$\}$. Find the minimum number of states required for DFA that accepts L ________.

(a) 4 (b) 5
(c) 6 (d) 7

Common Data (Q.33 and Q.34)

Consider the following ε-NFA:



Q.33 Find the cardinality of epsilon closure of state 'S' [ε-closure (q) is the set which contain all the states those are reachable from state q without reading any input symbol]

(a) 4 (b) 5
(c) 6 (d) 7

Q.34 Find the number of states required to construct an equivalent minimized DFA that accepts the same language as the given ε-NFA.

(a) 1 (b) 2
(c) 3 (d) None of these

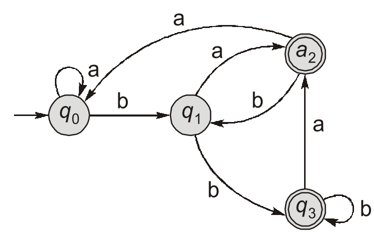
Q.35 Consider the following language:

$L_1 : \{a^n b^m \mid n > m$ or $n < m\}$
$L_2 : \{a^n b^m \mid n \geq m$ or $n \leq m\}$
$L_3 : \{a^n b^m \mid n > m$ and $n < m\}$
$L_4 : \{a^n b^m \mid n \geq m$ and $n \leq m\}$

Which of the following represent the regular language?

(a) L_4 and L_4 only
(b) L_2 and L_3 only
(c) L_2 and L_4 only
(d) L_2, L_3 and L_4 only

Q.36 Consider the following DFA 'D' given below:



Which of the following language represented by L(D)?

(a) $L(D) = \{w \in \Sigma^* \mid w$ is the string ending with b$\}$.
(b) $L(D) = \{w \in \Sigma^* \mid |w| \geq 2$, second-to last symbol w is b$\}$.
(c) $L(D) = \{w \in \Sigma^* \mid |w| \geq 2$, w is the string contain atleast one b$\}$.
(d) $L(D) = \{w \in \Sigma^* \mid w$ is the string not ending with "aa"$\}$.

Q.37 The number of DFA's with three states which can be constructed over the alphabet $\Sigma = \{a, b\}$ with designated initial state (fixed initial state) are ________.

Q.38 Which of the following is true?

(a) $\{(a^n b^n)^m \mid m, n \geq 0\} \equiv a^* b^*$
(b) $(a^* b)^* \equiv (a + b)^*$
(c) $\{a^n b^n \mid n \geq 1\} \equiv a^+ b^+$
(d) $\{a^{nk} \mid k > 0$, n is a positive integer$\} \equiv a^+$

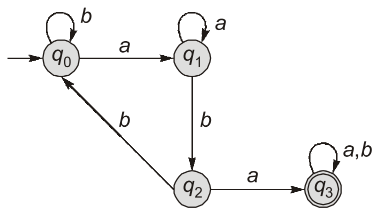
Q.39 The minimum number of state required for DFA construction for
$L(G) = \{w \mid w \in, \Sigma^*, n_a(w) \geq 2, n_b(w) \leq 1\}$ are ________.

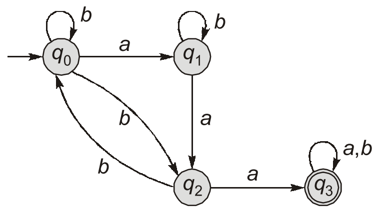
Q.40 Consider L be "the set of all binary strings whose last two symbols are different". The number of states needed in the minimum states deterministic finite automata accepting $\overline{L}$ is _______.

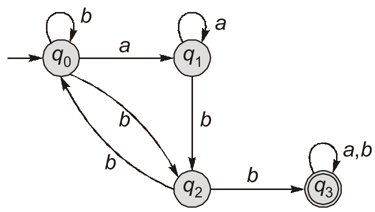
Q.41 The length of shortest string not present in language over $\Sigma = \{0, 1\}$ of the following regular expression is ________.

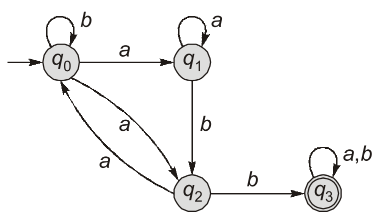
$$(10 + 0)^* (1 + 10)^*$$

Q.42 Let L be a language defined as $L = \{w \mid w$ contain 'aba' as substring, $w \in (a + b)^*\}$. Which of the following DFA will recognize the language L?

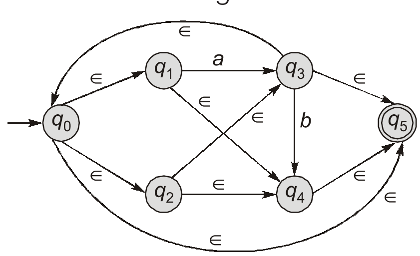
(a)


(b)


(c)


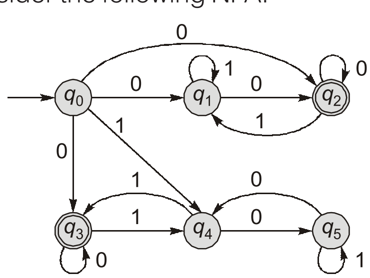
(d)


Q.43 Consider the following NFA.



Find the epsilon closure of state q_2.
(a) $\{q_4, q_3, q_5\}$ only
(b) $\{q_3, q_4\}$ only
(c) $\{q_0, q_5\}$ only
(d) $\{q_0, q_1, q_2, q_3, q_4, q_5\}$

Q.44 Consider the following NFA.



The above NFA accepts all those binary strings which represents the decimal numbers and are
(a) divisible by 6 only
(b) divisible by 2 and 3 only
(c) divisible by 2 or 3
(d) None of these

Q.45 In the minimal finite automata accepting the set of all strings over $\{a, b\}$ where the number of a's and the number of b's is divisible by 4 the total number of non-final states are ________.

Q.46 Find the minimum number of states in the DFA which accept the language of all strings that begin or end with 00 or 11.

(a) 6	(b) 7
(c) 8	(d) 9

Q.47 Consider the following language:

$$L = \{a^{m^n} \mid n \geq 1,\ m > n\}$$

Which of the following is correct regarding L?
(a) Language is CFL but not regular
(b) Language is CSL but not CFL
(c) Language is regular and the corresponding DFA has 3 states
(d) None of these

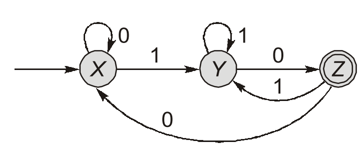
Q.48 Consider the following languages:

$L_1 = \{a^n \mid n \text{ is not a prime number}\}$
$L_2 = \{a^n \mid n = k^3 \text{ is for some } k \geq 0\}$
$L_3 = \{a^n \mid n = 2^k \text{ is for some } k \geq 0\}$

Which of the languages are regular?

(a) Only L_1 and L_2	(b) Only L_2
(c) Only L_2 and L_3	(d) None of these

Q.49 Consider the following DFA:



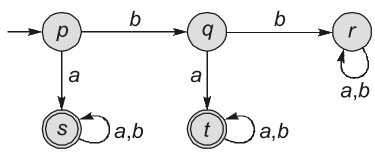
Which of the following regular grammar is equivalent to above DFA ?

(a) $X \to X0 \mid 0Y$	(b) $X \to 0X \mid 1Y$
$Y \to Y1 \mid Y0 \mid Z1$	$Y \to Y1 \mid Y0 \mid Z1$
$Z \to Z1 \mid 0X \mid 0 \mid 1 \mid \epsilon$	$Z \to Z1 \mid Z0 \mid 0 \mid 1 \mid \epsilon$

(c) $X \rightarrow 0X \mid 1Y$
$Y \rightarrow 1Y \mid 0Z$
$Z \rightarrow 1Y \mid 0X \mid 0 \mid 1 \mid \in$

(d) $X \rightarrow 0X \mid 1Y$
$Y \rightarrow 1Y \mid 0Z$
$Z \rightarrow 1Y \mid 0X \mid \in$

Q.50 Consider the DFA over alphabet $\Sigma = \{a, b\}$ is given below:



Consider the following statement about above DFA:

S_1 : Complement of $L(A)$ is context-sensitive.

S_2 : $L(A) = (a\,(a + b)^* + ba\,(a + b)^* + bb\,(a + b)^*)$.

S_3 : For the language accepted by A, the minimum DFA of A will contain 3 states.

S_4 : A accepts all strings over $\{a, b\}$ that ends with 'a'.

Which of the following are false?

(a) S_2 and S_3 only
(b) S_1 and S_3 only
(c) S_1 and S_2 only
(d) S_2, S_3 and S_4 only

Q.51 Let $L = \{baba\}$. Prefix is the set of all prefix string and suffix is the set of all suffix string over the language L is used to perform the following.

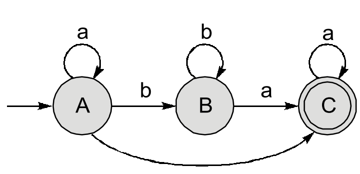
$$A = \{\text{Prefix}(L)\} \cap \{\text{Suffix}(L)\}$$

The number of strings exist in the language A are ______.

Q.52 Which of the following regular expressions describes the language over {0, 1} consisting of string 'w' where w starts with 0 and has odd length, or starts with 1 and has even length?

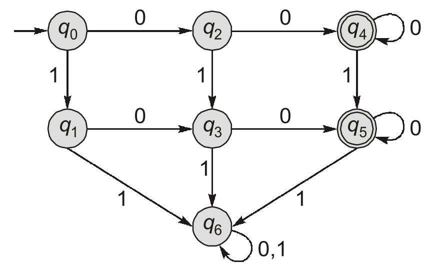
(a) $(0\,((0 + 1)\,(0 + 1))^* + 1\,(0 + 1))^*$
(b) $(0\,(0 + 1) + 1)\,(0 + 1)^*$
(c) $(0 + 1\,(0 + 1))\,((0 + 1)\,(0 + 1))^*$
(d) $0\,(0 + 1)^*\,(0 + 1)^* + 1\,(0 + 1)^*$

Q.53 What is the language accepted by the following NFA?



(a) $(a + b)^*a$
(b) $a^*\,b^*\,a^*$
(c) $a^*\,b^*\,a^*\,a$
(d) $(a + b)^+$

Q.54 Consider the machine M:



The language recognized by M is

(a) $\{w \in \{0, 1\}^* \mid$ every 1 in 'w' is followed by a 0$\}$
(b) $\{w \in \{0, 1\}^* \mid$ every string 'w' starts with '10' or '01' or '00'$\}$
(c) $\{w \in \{0, 1\}^* \mid w$ does not contain '11' as a substring$\}$
(d) $\{w \in \{0, 1\}^* \mid w$ contains at least two 0's and atmost one 1$\}$

Q.55 Consider alphabet $\Sigma = \{a, b\}$, the null/empty string λ and the set of strings X_0, X_1, X_2 and X_3 generated by the corresponding non-terminals of a regular grammar X_0, X_1, X_2 and X_3 are related as follows:

$$X_0 = \varepsilon + X_0\,b$$
$$X_1 = X_0\,a + X_1\,b + X_2\,a + X_3\,a$$
$$X_2 = X_1\,a + X_2\,b + X_3\,b$$

Which one of the following choice precisely represents the string in $X_1 \cup X_2$?

(a) $b^*a\,b^*a\,b^*a\,b^*$
(b) $(a + b)^*$
(c) $b^*\,a\,(a + b)^*$
(d) $b^*a\,b^*\,a\,b^*$

Answer Key:

1. (c)	**2.** (a)	**3.** (c)	**4.** (d)	**5.** (c)
6. (b)	**7.** (b)	**8.** (b)	**9.** (c)	**10.** (c)
11. (b)	**12.** (a)	**13.** (b)	**14.** (c)	**15.** (c)
16. (d)	**17.** (b)	**18.** (c)	**19.** (d)	**20.** (c)
21. (b)	**22.** (b)	**23.** (b)	**24.** (b)	**25.** (b)
26. (c)	**27.** (b)	**28.** (d)	**29.** (c)	**30.** (a)
31. (a)	**32.** (b)	**33.** (a)	**34.** (a)	**35.** (b)
36. (b)	**37.** (5832)	**38.** (d)	**39.** (7)	**40.** (5)
41. (4)	**42.** (a)	**43.** (d)	**44.** (c)	**45.** (15)
46. (c)	**47.** (c)	**48.** (d)	**49.** (d)	**50.** **(d)**
51. (3)	**52.** (c)	**53.** (c)	**54.** **(d)**	**55.** (c)

Student's Assignments 1 Explanations

1. (c)

R = $(a + \varepsilon)(bb^*a)^*$

R generates the language that do not contain two or more consecutive a's and do not end with b.

2. (a)

$1 \times 10 = 10 \Rightarrow 0^*1.0^*1010$

$[\because \text{dec}(1) = 0^*1, \text{dec}(10) = 0^*1010]$

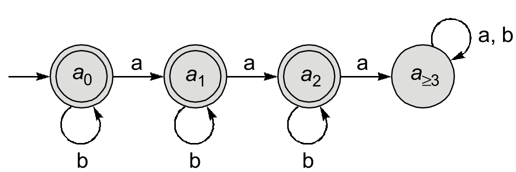
$10 \times 1 = 10 \Rightarrow 0^*1010.0^*1, 2 \times 5 = 10$

$\Rightarrow 0^*10.0^*101, 5 \times 2 = 10 \Rightarrow 0^*101.0^*10$

$\Rightarrow$ R.E. $= 0^*10^*1010 + 0^*1010\,0^*1 + 0^*100^*101 + 0^*1010^*10$

$\therefore$ L is regular language

3. (c)



All strings with atmost two a's are accepted by DFA.

4. (d)

Example 1

$L_1 = \{ba\}, L_2 = \{a^*b^*\}$

$L_1 \cup L_2 = ba + a^*b^*$

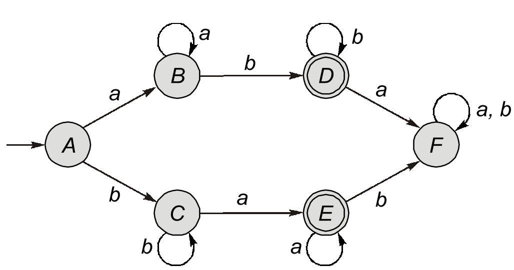
Example 2

$L_1 = \{ba\}, L_2 = \{a\}$

$L_1 \cup L_2 = ba + a$

$\therefore$ L_2 is regular but it may be finite or infinite.

5. (c)

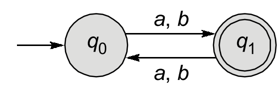


$\therefore$ 6 states are required to accept $a^+b^+ + b^+a^+$

6. (b)

Let $\Sigma = \{a, b\}$

Given, $\boxed{\delta(q_i, x) = q_{1-i}, x \in \Sigma, i = 0, 1}$



is equivalent DFA.

$\therefore$ DFA accepts all strings of odd lengths over Σ.

7. (b)

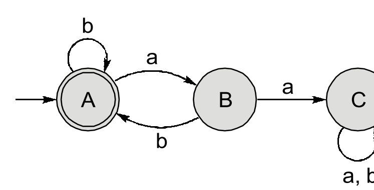
(a) Regular language: $1[(0 + 1)(0 + 1)]^*$

(b) Non regular language (Finding middle symbol is not possible)

(c) Regular language: $[(0 + 1)(0 + 1)]^*\,1$

8. (b)

Minimized DFA for the language R:



$[A] = (ab + b)^*$

$[B] = (ab + b)^*a$

$[C] = (ab + b)^*\,aa\,(a + b)^*$

$\therefore$ Three equivalence classes are present.

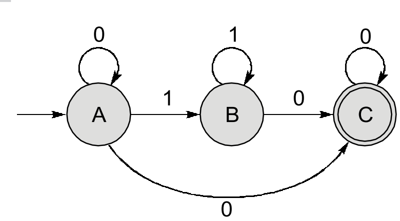
$([A] \cup [B] \cup [C] = \Sigma^*)$

9. (c)

$\bar{L}$ has every even length string and it contain all odd length strings which are not in the form of $w\,x\,w^R$.

L is non regular. Hence $\bar{L}$ is also non regular.

10. (c)



R.E. $= 0^*(11^*\,0 + 0)\,0^*$

$= 0^*((11^* + \varepsilon)\,0)\,0^* = 0^*\,1^*\,0\,0^*$

$= 0^*\,1^*\,0^*\,0$

So, option (c) is correct.